

Exploring Oracle Database 11g and 12c Partitioning New Features and Best Practices

Ami Aharonovich
ilOUG, DBAces

Keywords:

Oracle database 11g, Oracle database 12c, Partitioning New Features, Best Practices, Data Warehouse

Introduction

This session will explore new features and best practices for using various partitioning features while providing maximum performance and availability, including merge and exchange partitions and the latest database partitioning enhancements introduced in Oracle Database 11g and 12c, which take partitioning to perfection, and offer the widest choice of partitioning methods which include reference, interval, system, list, range, virtual column-based and new composite partitioning enhancements.

About the Speaker

Ami Aharonovich, Oracle ACE and Oracle Certified Professional (OCP) database developer and administrator, President of ilOUG (Israel Oracle User Group), Brillix-DBAces CEO, Oracle DBA consultant and instructor, specializes in providing professional services and delivering Oracle trainings dealing with Oracle database core technologies, frequent speaker at the Oracle Open World annual convention and various Oracle user group events and conferences around the globe.

About Brillix-DBAces

The company provides expert consulting and training services in database development and administration, offering complete end-to-end integrative solutions and services for customer's database environments including:

- Database architecture, design, implementation and maintenance
- Database troubleshooting, performance monitoring and tuning
- Designing and programming advanced database applications
- Designing and implementing advanced solutions for high availability and replication environments
- Providing remote database advanced services
- Delivering professional database trainings and hands-on seminars

Oracle Partitioning Basics

Oracle partitioning feature was first introduced in Oracle 8.0 in 1997 and since then has become one of the most important and successful functionalities of the Oracle database, a key tool for building large systems with extreme high performance and availability requirements. Partitioning enhances database manageability, performance and availability for a wide variety of applications. It allows tables and indexes to be subdivided into smaller more manageable pieces called partitions or even sub-partitions. Each piece is a different database segment and can be managed individually and function independently of others.

Partitioning Strategies – Single Level

Oracle partitioning offers the following strategies for data distribution:

- Range (Oracle 8): Maps data to partitions based on ranges of partition key values for each partition
- Hash (Oracle 8i): Maps data to partitions by using a hashing algorithm applied to a partitioning key
- List (Oracle 9i): Maps data to partitions by using a list of discrete values for the partitioning column
- Interval (Oracle 11g): Maps data to partitions of ranges that are automatically created following a specified interval
- System (Oracle 11g): Allows the application to explicitly map rows to arbitrary partitions

Composite Partitioning Techniques

Composite partitioning is when a table is partitioned by one data distribution method and then each partition is further subdivided into sub-partitions using a second data distribution method. All sub-partitions for a given partition together represent a logical subset of the data. The following table illustrates the wide range of available composite partitioning techniques across different Oracle database versions:

1 st Level Partitioning	2 nd Level Partitioning	Oracle Database Version
Range	Hash	Oracle 8i
	List	Oracle 9i
	Range	Oracle 11g
List	Range	Oracle 11g
	List	Oracle 11g
	Hash	Oracle 11g
Hash	Range	Oracle 11g
	List	Oracle 11g
	Hash	Oracle 11g
Interval	Range	Oracle 11g
	List	Oracle 11g
	Hash	Oracle 11g

Partitioning Benefits

Partitioning provides tremendous benefits to a wide variety of applications by improving performance, manageability, and availability. It is common for partitioning to improve the performance of certain queries or maintenance operations by an order of magnitude. Moreover, partitioning can greatly simplify common administration tasks. Partitioning is a key tool for building large database systems or systems with extremely high availability requirements. Partitioning benefits include:

- Better database manageability and availability: Different partitions that belong to the same table/index can reside in different tablespaces, have distinct storage clauses and be maintained by granular commands

- Partitions are transparent to existing applications and do not require any code changes

Partitioning provides a number of performance benefits including:

- Oracle database optimizer automatically eliminates partitions that do not need to be scanned (partition pruning)
- Join operations can be optimized to perform the join “by the partition” (partition-wise join)

Partitioning Best Practices

- Exchanging partitions: You can convert a partition or a sub-partition into a non-partitioned table, and a non-partitioned table into a partition or sub-partition of a partitioned table by simply using the exchange partition command which will cause exchanging the database segments. This is one of the best and most powerful features in Oracle database partitioning and it gives the ability to load data quickly and easily with minimal impact on current application users.
- Using compression: For partitioned tables and indexes you can compress some or all partitions using Oracle database compression feature. The compression attribute can be declared for either the tablespace, table or partition level. For indexes, using key compression eliminates repeated occurrences of key column prefix values.
- Copy statistics: Very important and useful feature available since Oracle database version 10.2.0.4. This feature provides the ability to simply copy statistics of the source partition or sub-partition to the destination partition or sub-partition thus saving significant expensive time avoiding gathering statistics on actual database segments.
- Sub-partition templates: Creating sub-partitions in a composite partitioned table can be done using a sub-partition template. The template simplifies the specification of sub-partitions by not requiring that a sub-partition descriptor will be specified for each and every sub-partition defined in the table. Instead, you can describe sub-partitions only once in the template, and then apply the template to every partition in the table.

Oracle Database 11g Partitioning New Features

- Interval partitioning: Can be used to automate the creation of range partitions. Oracle will create any partition automatically as needed whenever data for a partition is inserted for the very first time. This feature greatly improves the manageability of a ranged partitioned table. Available techniques are interval, interval-list, interval-hash, interval-range and interval-reference (available only in Oracle database version 12c). When using interval partitioning, you must specify at least one range partition when creating the table. In an interval partitioning table, the partitioning key column must be of NUMBER or DATE data type.
- Virtual column-based partitioning: Allows partitioning key to be defined by an expression, using one or more existing columns of a table and storing the expression as metadata only. This new feature enables a more comprehensive match various business requirements. It is supported with all basic partitioning strategies, and it can also be used with interval partitioning as well as the partitioning key for a referenced partitioned table. Virtual columns are treated as regular real columns except for the fact that no DML operations are allowed on.

- Reference partitioning: Allows partitioning a table by leveraging existing parent-child relationship, meaning that the partitioning strategy of the parent table is inherited to its child table without the necessity to store parent's partitioning key column in the child table. The child table will transparently inherit all partitioning maintenance operations from the parent table to the child table. This feature automatically enables partition-wise joins for the equal-partitions of the parent and child tables and it is perfect for use in case of a star schema in data warehouses; partition the fact table according to the dimension table.
- System partitioning: Enables application-controlled partitioning, meaning it allows the application to explicitly map rows to arbitrary partitions without defining any partitioning rule. It provides the ability to break down a table into meaningless partitions while all aspects of partitioning are manually controlled by the application. This means that common performance benefits of partitioned tables are not available (as there is no partitioning key) and so it does not provide any support for traditional partition pruning, partition wise joins, and so on.

Oracle Database 12c Partitioning New Features

- Partial indexes for partitioned tables: This new feature allows defining new index attributes which are only applicable to indexes on partitioned tables, meaning that indexes can be created on a subset of the partitions of a table (instead on all partitions of a table). It provides more flexibility in index creation for partitioned tables. For example, you can choose not to index the most recent partition to avoid any index maintenance work at data insertion time, therefore maximizing data load speed.
- ONLINE move operations: Since Oracle database version 12c, partition maintenance operations can now be done online, allowing DMLs to occur while data maintenance operation is in process. The ALTER TABLE ... MOVE PARTITION command has now become a non-blocking online DDL statement while allowing DML operations to continue running uninterrupted on the partition that is being moved. In addition, all global indexes are maintained during the move partition, so that a manual index rebuild operation is no longer required. Using this new ONLINE clause, will cause the move operation to wait for existing running transactions to be finished (commit or rollback) and is then it will be executed.
- Partition maintenance operations on multiple partitions: Means that starting from Oracle database version 12c partition maintenance operations can now be performed on multiple partitions as part of a single partition maintenance operation (one atomic operation). This feature greatly simplifies application development and leads to more efficient partition maintenance using less system resources.
- Asynchronous global index maintenance for DROP and TRUNCATE partition commands: Global index maintenance operation is now decoupled from the DROP and TRUNCATE partition maintenance operations without rendering a global index to become unusable. The index maintenance operation will be automatically performed asynchronously and can be delayed to a later point in time.
- Interval reference partitioning: Using referenced partitioned table can now be combined with using interval partitioning as the top partitioning strategy. Using this new feature will significantly enhance Oracle's partitioning capabilities to model the database schema according to real business needs.

Oracle white paper: Partitioning with Oracle Database 12c:

<http://www.oracle.com/technetwork/database/options/partitioning/partitioning-wp-12c-1896137.pdf>

Contact address:

Ami Aharonovich

Brillix-DBAces

2/20 Hazamir St.

Kiryat Ono

Israel 5550702

Phone: +972(524)377737
Fax: +972(77)3373770
Email: Ami@DBAces.com
Internet: www.DBAces.com